Internet of Things

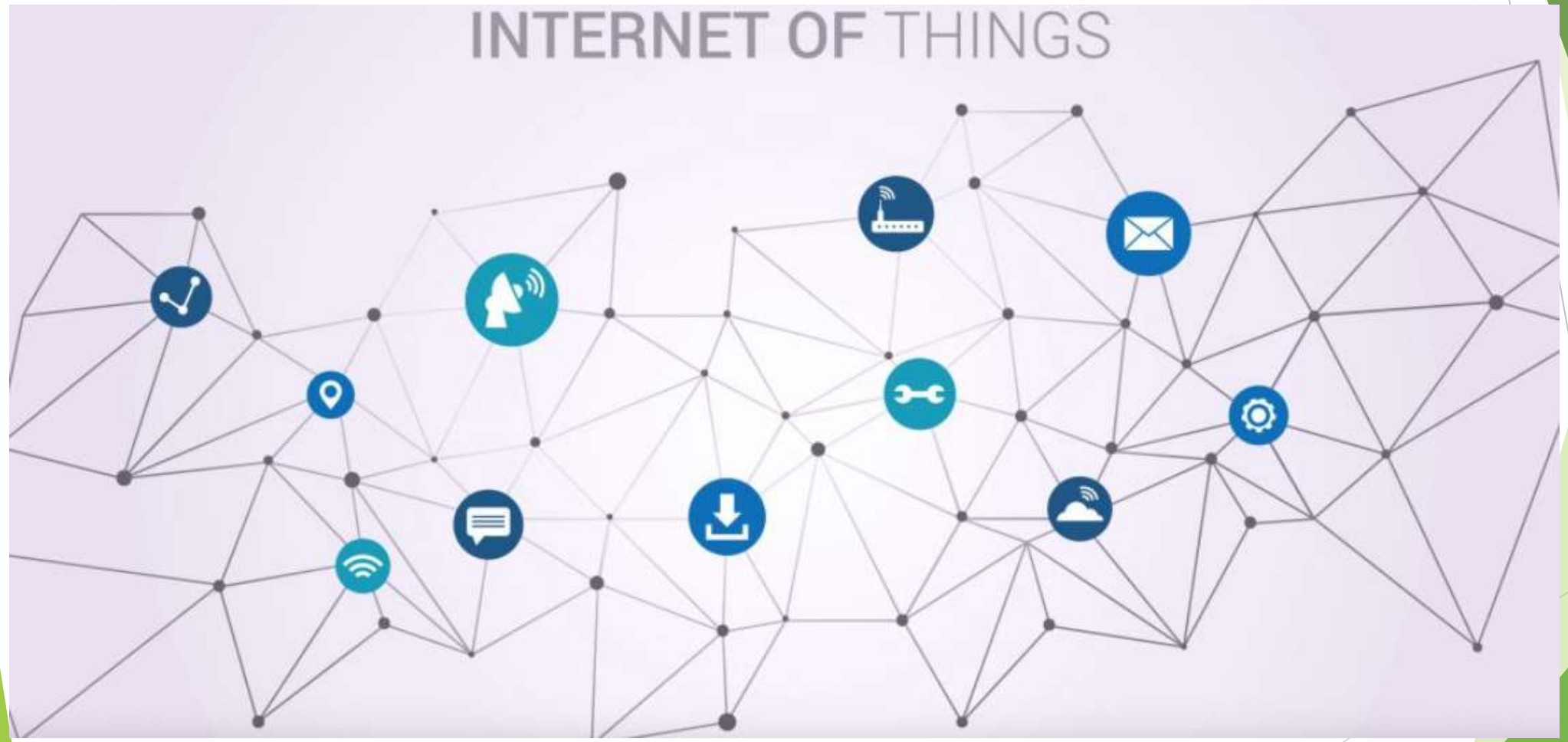

Top IoT Trends in 2020
Smart Cities
Smart Factories
Data-Driven Healthcare
Artificial Intelligence (AI)
Edge Computing
Smart Retail
Fitness
Telecom-munications

What is IOT?

IoT is a new revolution to the internet due to the advancement in sensor networks, mobile devices, wireless communication, networking and cloud technologies.

WHY?

- IoT is a dynamic global network infrastructure of physical and virtual objects having unique identities,
- which are embedded with software, sensors, actuators, electronic and network connectivity
- to facilitate intelligent applications by collecting and exchanging data.


INTERNET OF THINGS

Goal?

- The main goal of IoT is to configure, control and network the devices or things, to internet, which are traditionally not associated with the internet

i.e thermostats, utility meters, a Bluetooth connected headset, irrigation pumps and sensors or control circuits for an electric car's engine that make energy, logistics, industrial control, retail, agriculture and many other domain smarter.

Characteristics of IoT:

- Various characteristics of IoT are:
- • Dynamic and self-adapting
- • Self-configuring
- • Interoperable Communication protocols
- • Unique identity
- • Integrated into information network

Dynamic and self-adapting:

The IoT devices can dynamically adapt with sensed environment, their operating conditions, and user's context and take actions accordingly. For ex: Surveillance System.

Self-configuring:

I. IoT devices can be able to upgrade the software with minimal intervention of user, whenever they are connected to the internet.

II. They can also setup the network i.e a new device can be easily added to the existing network. For ex: Whenever there will be free wifi access one device can be connected easily.

Interoperable Communication:

- IoT allows different devices (different in architecture) to communicate with each other as well as with different network. For ex: MI Phone is able to control the smart AC and smart TV of different manufacturer

Unique identities:

- The devices which are connected to the internet have unique identities i.e IP address through which they can be identified throughout the network.
- The IoT devices have intelligent interfaces which allow communicating with users. It adapts to the environmental contexts.
- It also allows the user to query the devices, monitor their status, and control them remotely, in association with the control, configuration and management infrastructure.

Integrated into information network:

- The IoT devices are connected to the network to share some information with other connected devices. The devices can be discovered dynamically in the network by other devices.

For ex. If a device has wifi connectivity then that will be shown to other nearby devices having wifi connectivity.

- The devices ssid will be visible though out the network. Due to these things the network is also called as information network.
- The IoT devices become smarter due to the collective intelligence of the individual devices in collaboration with the information network. For Ex: weather monitoring system. Here the information collected from different monitoring nodes (sensors, arduino devices) can be aggregated and analysed to predict the weather

Physical Design of IoT:

The physical design of an IoT system is referred to the Things/Devices and protocols that used to build an IoT system. all these things/Devices are called Node Devices and every device has a unique identity that performs remote sensing, actuating, and monitoring work. and the protocols that used to established communication between the Node devices and server over the internet.

Physical Design of IoT

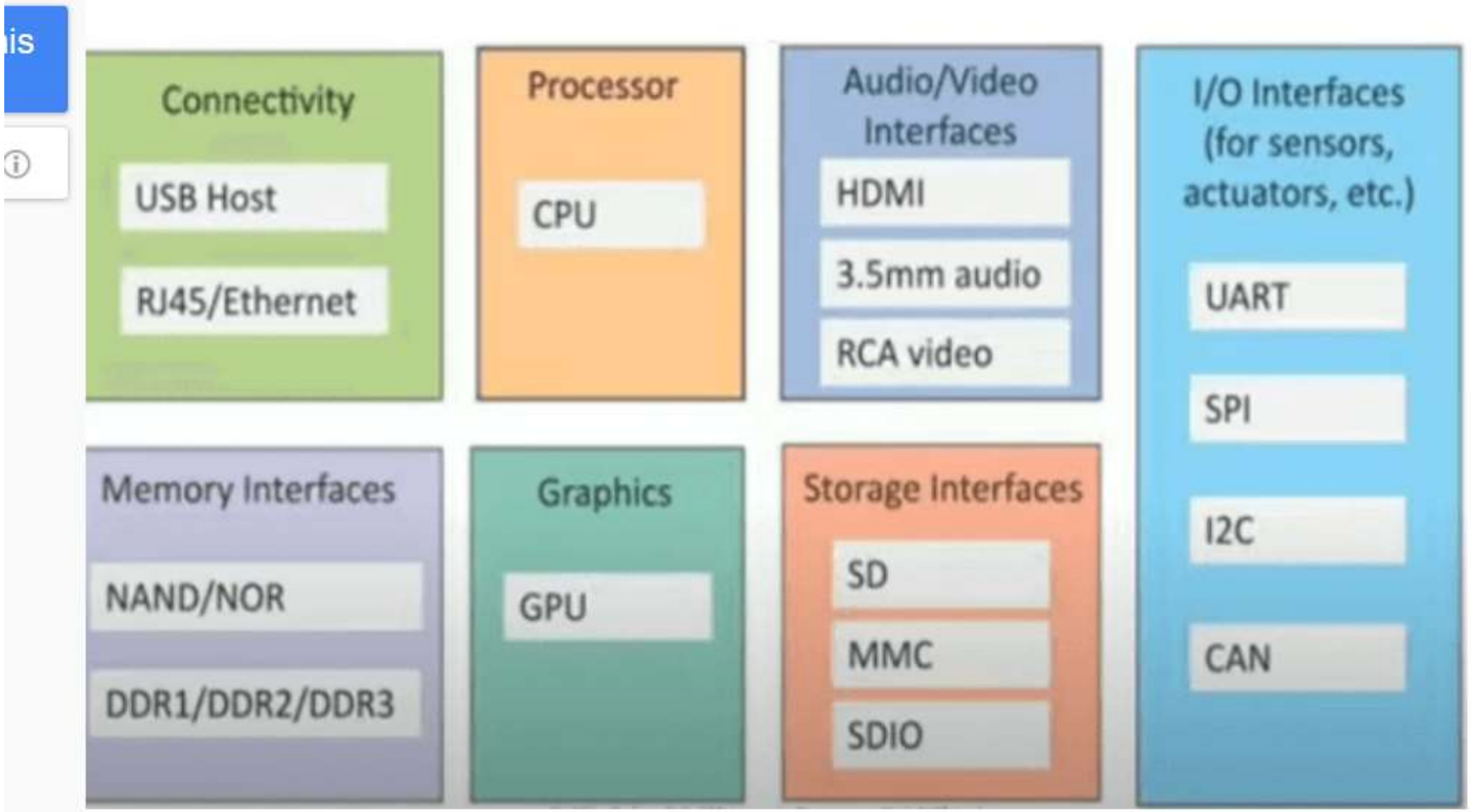
Things

Protocols

Physical Design of IoT:

Things in IoT:

- I. IoT i.e Internet of things, where things refer to the IoT devices which have unique identities and can perform remote sensing, actuating and monitoring capabilities (ex: combination of sensors, actuators, Arduino, relay, non IoT devices).
- II. The IoT devices can share information with as well as collect information from other connected devices and applications (directly and indirectly).
- III. They can process the data locally or in the cloud to find greater insights and put them into action based on temporal and space constraints (i.e space memory, processing capabilities, communication latencies and speeds and deadlines).
- IV. IoT devices can be of varied types. For ex: wearable sensors, smart watches, LED lights, automobiles and industrial machines.


Connectivity
USB Host
RJ45/Ethernet
Processor
CPU
Audio/Video Interfaces
HDMI
3.5mm audio
RCA video
I/O Interfaces (for sensors, actuators, etc.)
UART
SPI
I2C
CAN
Memory Interfaces
NAND/NOR
DDR1/DDR2/DDR3
Graphics
GPU
Storage Interfaces
SD
MMC
SDIO

Logical design of IoT:

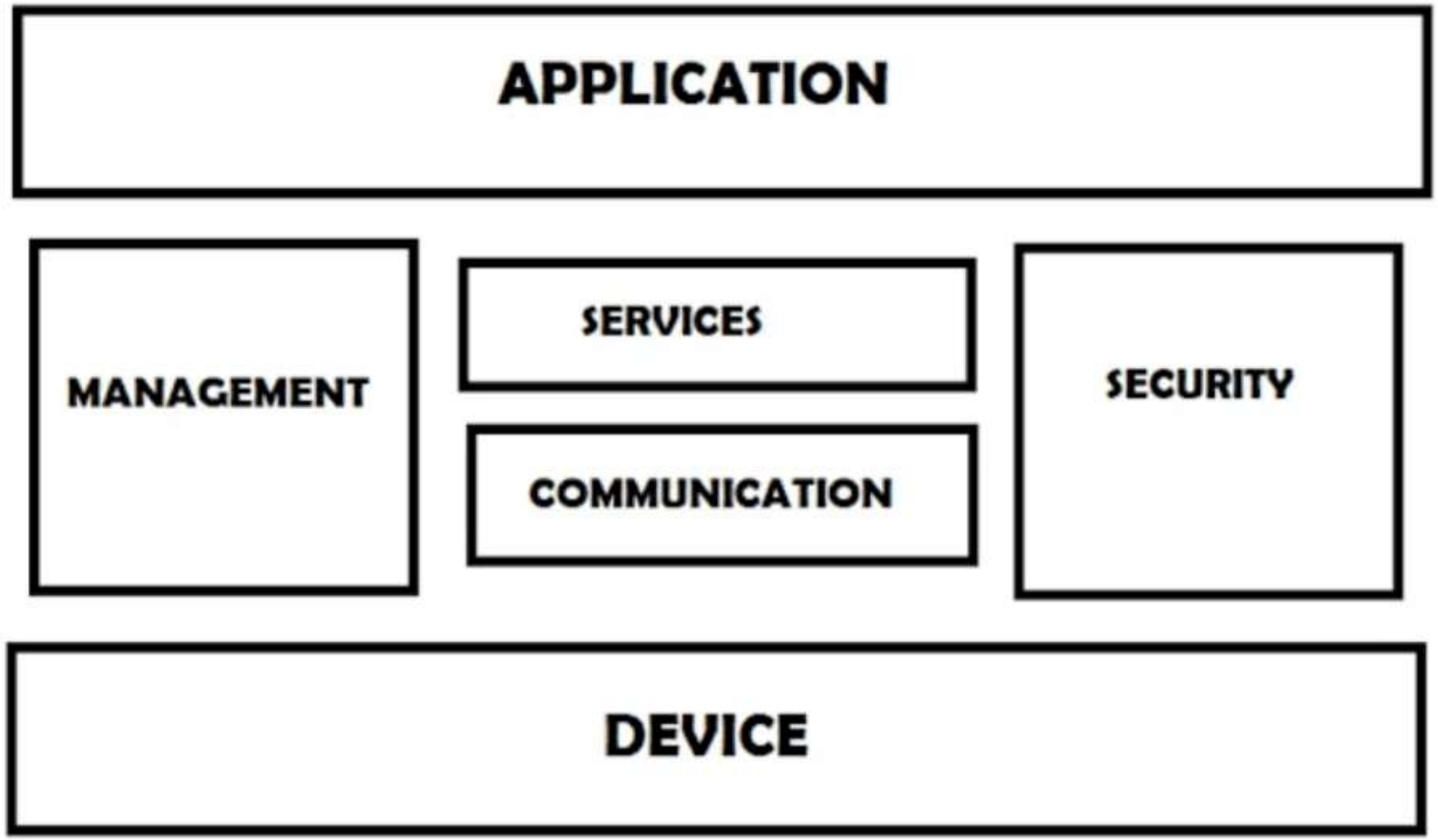
- Logical design of IoT refers to an abstract representation of entities and the processes without going into the details of the implementations. The logical design includes functional block of IoT and the communication APIs
- **The logical design** of an IoT system refers to an abstract representation of entities and processes without going into the low-level specifies of implementation. it uses **Functional Blocks**, **Communication Models**, and **Communication APIs** to implement a system.

Logical Design of IoT

- IoT Functional Blocks
- IoT Communication Models
- IoT Communication APIs

IoT Functional blocks

- An IoT system consist number of functional blocks like Devices, services, communication, security, and application that provides the capability for sensing, actuation, identification, communication, and management.
- These functional blocks consist of devices that provide monitoring control functions, handle communication between host and server, manage the transfer of data, secure the system using authentication and other functions, and interface to control and monitor various terms.


APPLICATION
MANAGEMENT
SERVICES
COMMUNICATION
SECURITY
DEVICE

Application
It is an interface that provides a control system that use by users to view the status and analyze of system.

Management
This functional block provides various functions that are used to manage an IoT system.

Services
This functional block provides some services like monitoring and controlling a device and publishing and deleting the data and restore the system.

Communication
This block handles the communication between the client and cloud-based server and sends/receives the data using protocols.

Security
This block is used to secure an IoT system using some functions like authorization, data security, authentication, 2 step verification, etc.

Device
These devices are used to provide sensing and monitoring control functions that collect the data from the outer environment.

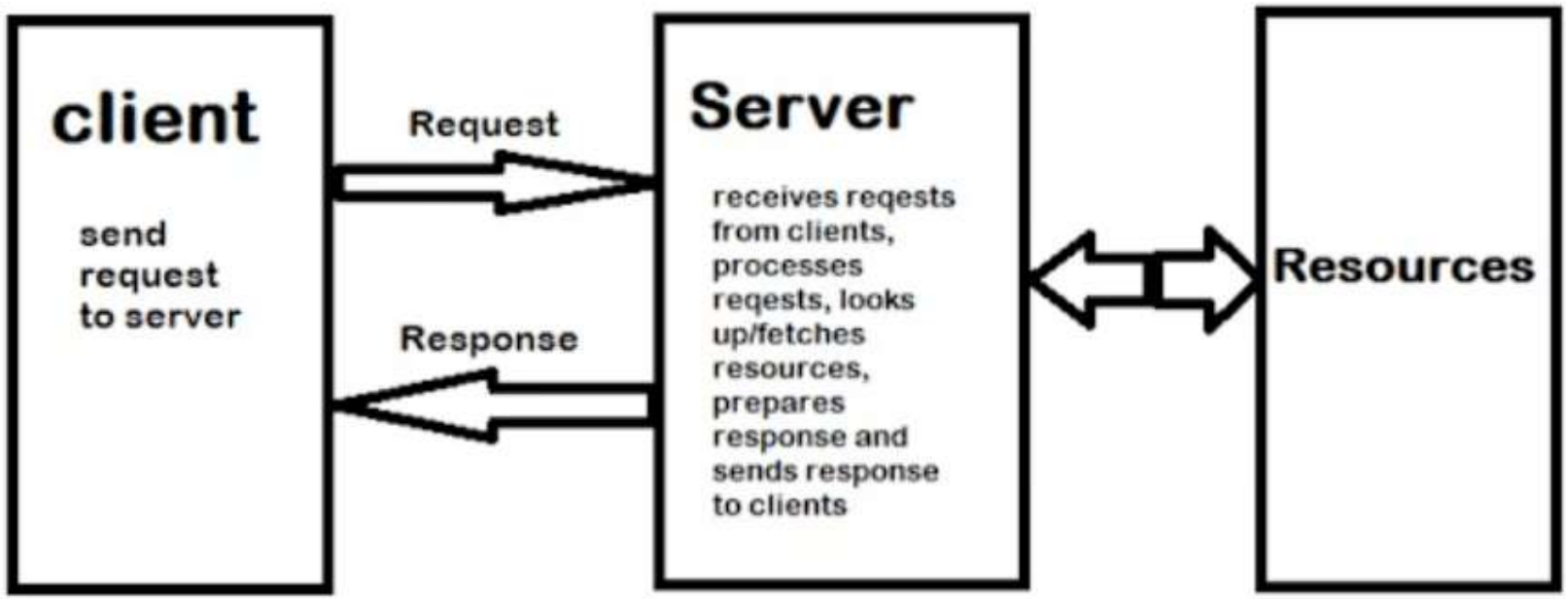
IoT Communication Models

- There are several different types of models available in an IoT system that used to communicate between the system and server like the

1. request-response model,
2. publish-subscribe model,
3. push-pull model,
4. exclusive pair model, etc

Request-Response Communication Model

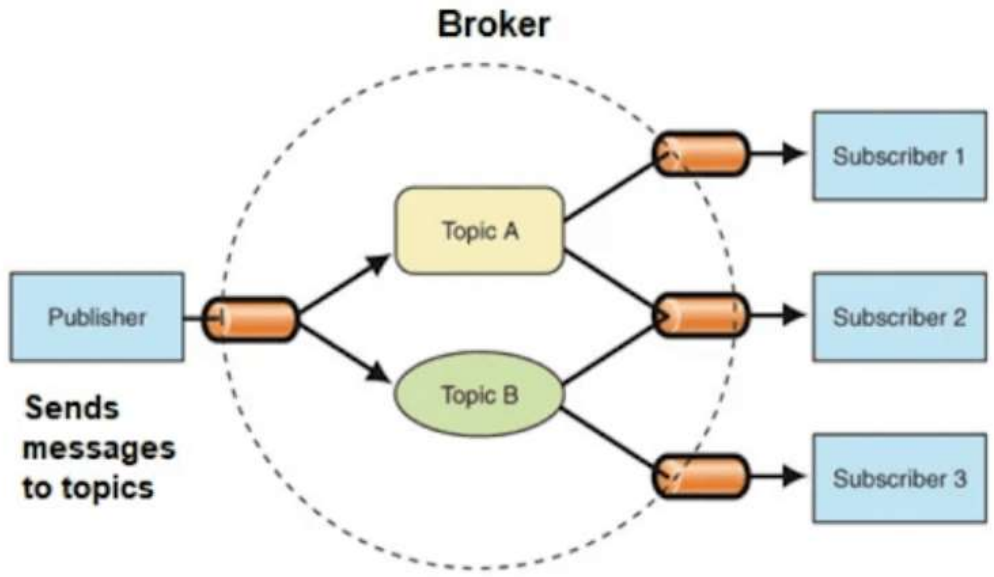
This model is a communication model in which a client sends the request for data to the server and the server responds according to the request. when a server receives a request it fetches the data, retrieves the resources and prepares the response, and then sends the data back to the client. HTTP works as a request-response protocol between a client and server.



Request-Response Communication Model

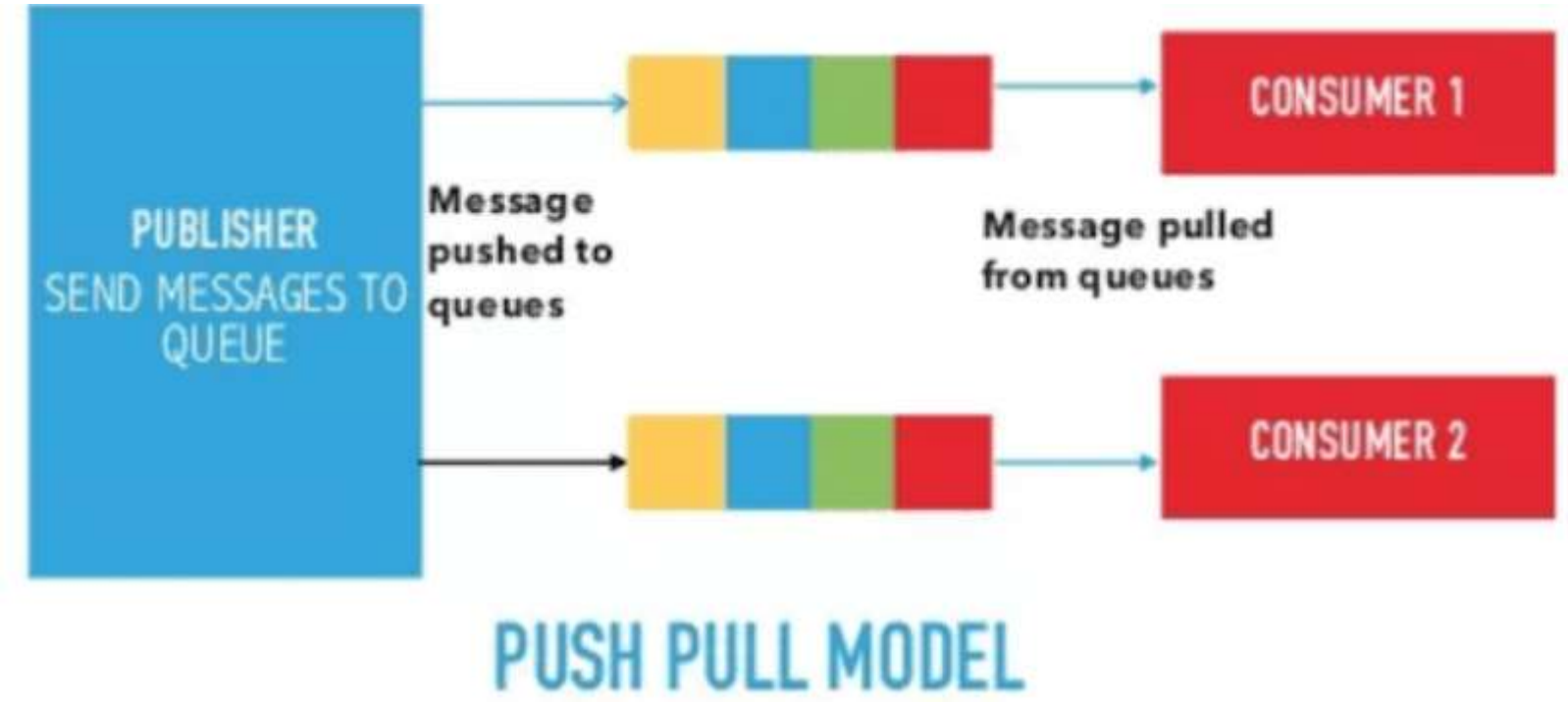
Publish-Subscribe Communication Model

In this communication model, broker between publisher and consumer exist. Here publishers are the source of data but they are not aware of consumers. they send the data managed by the brokers and when a consumer subscribes to a topic that managed by the broker and when the broker receives data from the publisher it sends the data to all the subscribed consumers.



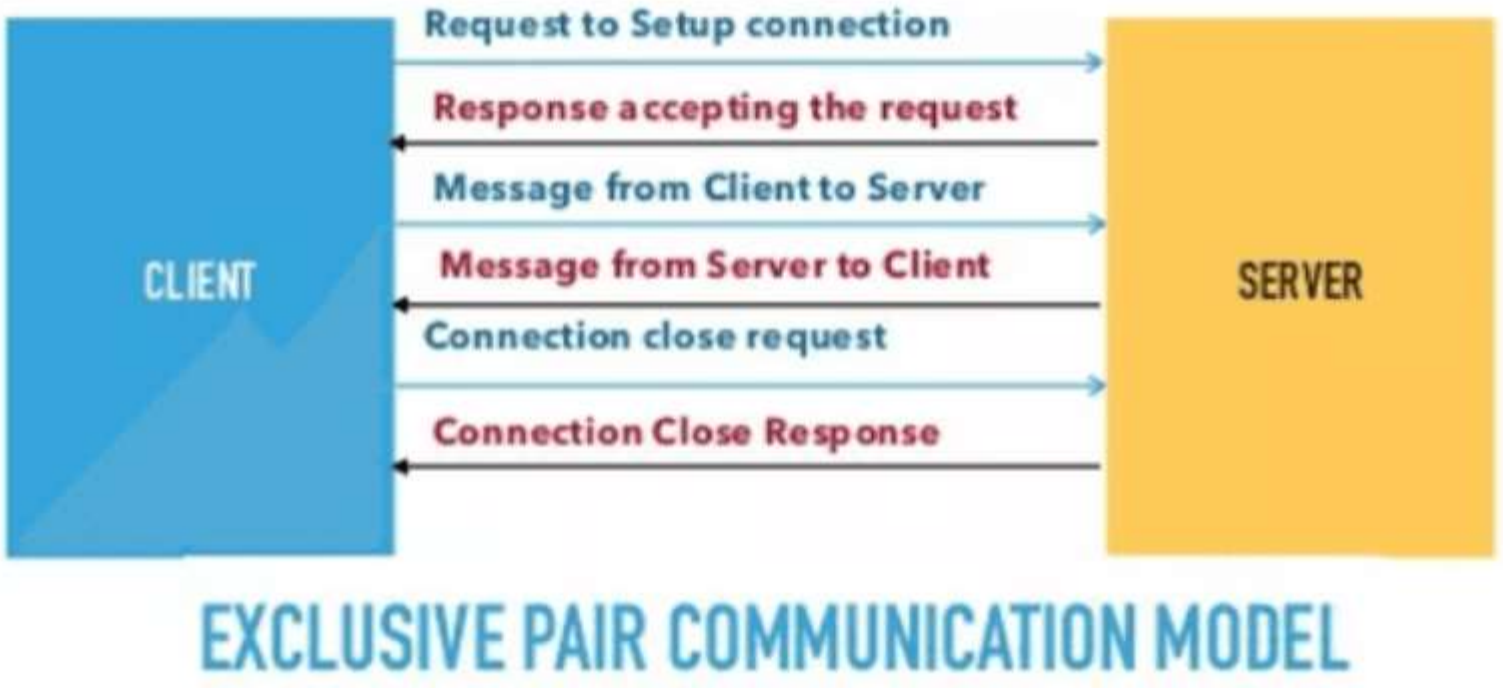
Push-Pull Communication Model

- It is a communication model in which the data push by the producers in a queue and the consumers pull the data from the queues. here also producers are not aware of the consumers.



Exclusive Pair Communication Model

It is a bidirectional fully duplex communication model that uses a persistent connection between the client and server. here first set up a connection between the client and the server and remains open until the client sends a close connection request to the server.



IoT communication APIs

These APIs like REST and WebSocket are used to communicate between the server and system in IoT.

REST-based communication APIs

- Representational state transfer(REST) API uses a set of architectural principles that used to design web services. these APIs focus on the systems' resources that how resource states are transferred using the request-response communication model.

REST-based communication APIs

- It helps to design web services and web APIs that focus on a system's resource and how resource states are addressed and transferred.
- It follows Request-response communication model and unidirectional communication for request. The clients send request to URIs using methods defined by the HTTP protocols (GET, PUT, POST, DELETE).
- RESTful web service is a "web API" implemented using HTTP and REST principle. RESTful web service is a collection of resources which are represented by URIs.
- RESTful Web services can support various internet media types (JSON, XML).

This API uses some architectural constraints.

- Client-server
- Stateless
- Cache-able
- Layered System
- Uniform interface
- Code on demand
- Scalability

Client-server

- The client is not aware of the storage of data as it is concerned about the server and similarly the server should not be concerned about the user interface as it is a concern of the client.
- This support independent development and updating of server and client. no matter how the client is using the response of the server and no matter how the server is using the request of the client.

Stateless

- It means each request from the client to the server must contain all the necessary information to understand by the server.
- if the server can't understand the request of the client then it can't fetch the request data in a proper manner.

Cacheable

- In response, if the cache constraints are given then a client can reuse that response l
- n a later request. it improves the efficiency and scalability of the system without loading the extra data.
- A RESTful web APIs is implemented using HTTP and REST principles

- Layered System: This constraint limits the behaviour of components i.e each component cannot see beyond the immediate layer with which they are interacting. Ex: client cannot say whether it is connected directly to the end server or to an intermediary. It improves scalability by allowing intermediaries to respond to requests instead of the end server without the client having to do anything different.
- Uniform interface: The method of communication between a client and a server must be uniform
- Code on demand: Servers can provide executable codes or scripts for clients to execute in their context.
- Scalability: it supports both horizontal and vertical scalability. As it is stateless so scalability is easier to implement.

WebSocket based communication API

- Websocket API helps to design web services and web APIs.
- It allows bi-directional, full-duplex communication between clients and servers.
- It follows the exclusive pair communication model.
- It supports stateful protocols. It does not require a new connection to be set up for each message to be sent. There is no overload for connection set up and termination request for each message. So Web socket API reduces the network traffic and latency.
- It is suitable for IoT applications that have low latency or high throughput requirements.

Some Difference……

REST	Websocket
. It supports Request-response communication model.	It supports Exclusive-pair communication model.
It supports stateless protocol	It supports stateful protocol.
It supports unidirectional communication between client and server as only client can send request to server and server only respond to the request.	It supports bidirectional communication between client and server i.e client and server both can request to each other.
It is half duplex	It is full duplex
It uses multiple TCP connection for each search over HTTP.	It uses single TCP connection for search over HTTP
Since it does not store the request information so each time it needs to provide all the information while creating communication with server. For this reason header overhead increases	Header overhead is less
It supports both horizontal and vertical scalability.	Vertical scaling is easier than horizontal scaling.

IoT enabling technologies:

- Wireless sensor networks
- Cloud computing
- Big Data analytics
- Embedded systems
- Communication protocols

Wireless Sensor networks:

- Wireless sensor network comprises of distributed devices, wireless sensors. These devices with sensors are used to monitor the environment and physical conditions. Since all the nodes are wireless so they communicate with each other through wifi or Bluetooth.
- It consists of several end nodes and routers as well as coordinator.
- Sensors are attached with end nodes. Each router can also be called as end node.
- Routers are responsible for routing the data packets from end nodes to the coordinator nodes. Coordinator node connects the WSN to the internet. The Coordinator node can be another arduino, raspberry pi or any other IoT DIY device.
- It collects the data from all the nodes.
- WSNs are enabled by wireless communication protocols such as IEEE802.15.4.

Cloud Computing:

- It is an emerging technology which enables on-demand network access to computing resources like network servers, storage, applications and services that can be rapidly provisioned and released.
- On demand: we invoke cloud services only when we need them, they are not permanent part of IT infrastructure.
- Pay as you go model: You pay for the cloud services when you use them, either for the short period of time or longer duration (for cloud based storage).
- Cloud provides various services such as

i. IAAS: Infrastructure as a service
ii. PAAS: Platform as a service
iii. SAAS: Software as a service

IAAS

- Instead of creating a server room we will hire it from a cloud service provider.
- Here user will not use its local computer, storage and processing resources rather it will use virtual machine and virtual storage, servers, networking of third party.

- Here the client can deploy the OS (operating system), application of his own choice.
- User can start, stop, configure and manage the virtual machine instances and virtual storage.

PAAS:

- User can develop and deploy applications. For ex. We are using various online editors to write codes like online arduino IDE, C IDE, APIs, software libraries.
- Here we don’t need to install anything. The cloud service provider will manage servers, network, OS and storage.
- The users will develop, deploy, configure and manage applications on the cloud infrastructure

SAAS:

- It provides complete software application or the user interface to the application itself.
- The user is not concerned about the underlying architecture of cloud only service provider is responsible for this.
- It is platform independent and can be accessed from various client devices such as workstation, laptop, tablet and smart phone, running different OS. Ex: The online software we use like online image converter, doc converter etc.

Big data analytics:

Big data refers to large amount of data which cannot be stored, processed and analysed using traditional database like (oracle, mysql) and traditional processing tools. In big data analytics BIG refers to 5 Vs.

- Volume
- Velocity
- Variety
- Veracity
- Value

- Volume: volume refers to the massive amount of data generated from the IoT systems. There is no threshold value for generated data. It is difficult to store, process and analyse using traditional database and processing tools.

Ex: The volume of data generated by modern IT, industrial and healthcare system.

- Velocity: The rate at which the data is generated from the IoT system. This is the primary reason for the exponential growth of data. Velocity refers to how fast the data is generated and how frequently it varies.
- Ex: Modern IT, industrial and other systems like social networking sites are generating data at increasingly higher speed.

- Variety: Variety refers to different forms of data. Since there are various domain of IoT so various type of data are generated from different IoT domain. Those data is called as sparse data. Those data include text, audio, video etc.. The variety of data is mainly divided into 3 types i.e. ✓ structured ✓ semi structured ✓ unstructured
  1. Structured data: The data which has a fixed format to be stored is known as structured data. The data stored in database like oracle, MySQL is an example of structured data. With a simple query data can be retrieved from the database.
  2. Semi-structured data: The data which has not a fixed format to be stored but uses some elements and components through which they can be analysed easily is known as semi structured data. Ex: HTML, XML, JSON data
  3. Unstructured data: The data which has not any fixed format. It is difficult to store and analyse. It can be analysed after converting into structured data. Ex: Audio, video (gif, audio with lyrics), Text (containing special symbols).
- Veracity: The data in doubt is known as veracity. Sometimes what happen it is very difficult accept the data stored in database. This happens due to typical error, corrupted storage or data.
- Value: It is efficient to access big data if we can turn it into values i.e we can find greater insights from it so that we can perform some action to get the desired output. This will be beneficial for the organisation. Otherwise it has no use.

Embedded Systems:

- An embedded system is a computer system that has hardware and software embedded to perform specific task.
- The key components of an embedded system include microprocessor or micro controller, memory (RAM, ROM, Cache), networking units (Ethernet, Wi-Fi adapter), input/output units (display, keyboard, etc) and storage (flash memory). They use some special types of processor such as digital signal processor, graphics processor and application specific processor). Embedded system uses embedded OS like RTOS. 3.
- Ex. Of embedded systems: digital watch, digital camera, vending machines.

Communication protocols:

- Protocol is nothing but rules and regulations. Communication protocol is the backbone of the IoT system.
- It allows interoperability among various devices. It enables network connectivity and coupling to applications.
- It allows devices to exchange data over the network. These protocols define data exchange format, data encoding, addressing schemes for devices and routing of packets from source to destination. It also includes sequence control, flow control and retransmission of lost packets.

IoT levels:

- Based upon the number of monitoring nodes used, type of data base used, complexity/ simplicity of analysis, computation there are 6 levels of IoT. Different applications are implemented based on this level. The IoT systems consist of these following components.

A. Device
B. Resources
C. Controller Service
D. Database
E. Web Service
F. Analysis Component
G. Application

Device: The IoT device allows identification, remote sensing, actuating, and remote monitoring capabilities.

Resource: Resources are the **software components on the IoT device** for accessing, processing and storing sensor information, or controlling actuators connected to the device Resources include the software components that enable network access for the device. For ex: The programs that we have written for object detection using IR sensor, to find out the distance using ultra sonic sensor etc.

Controller Service: Controller service is a native service that runs on the device and interacts with the web services. Controller service sends data from the device to the web service and receives commands from the application for controlling the web services. For ex: The ESP 8266 programming, setting of API keys, SSID etc. .
Suchismita Mohanty

Database: Database can be either local or in the cloud and stores the data generated by the IoT device.

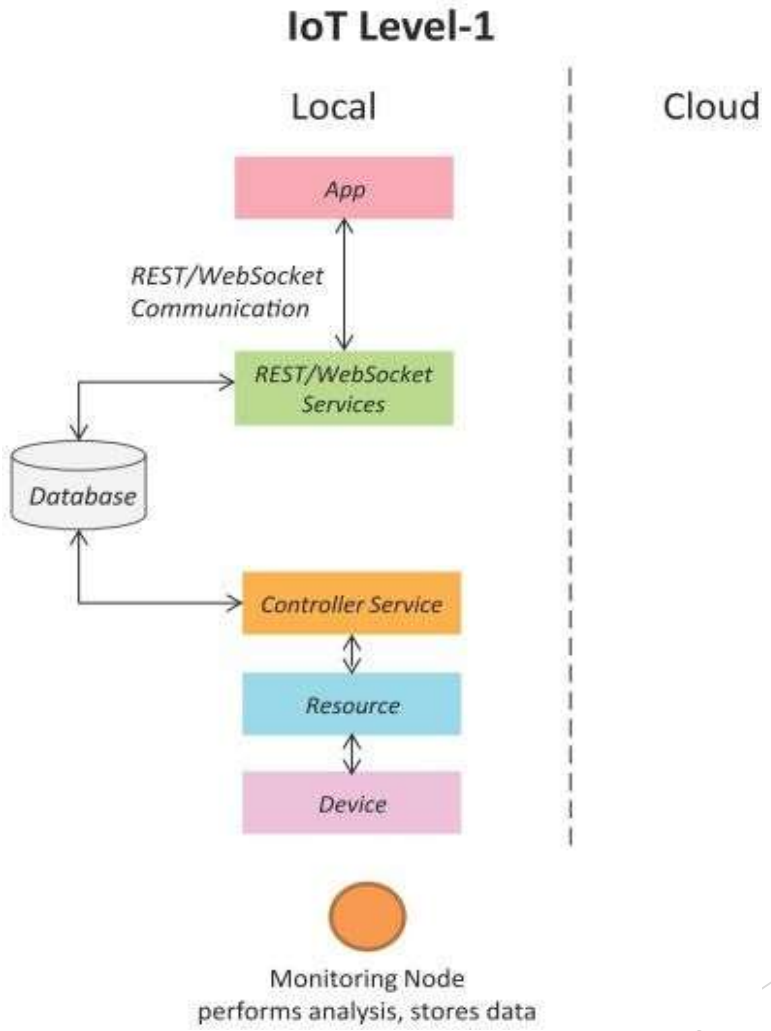
- Web Service: This act as an interface between IoT device, application, database and analysis components. Web services can be implemented using HTTP and REST principle or using WebSocket protocol.

- Analysis Component: The analysis component is responsible for analysing the IoT data and generates results inform which are easy for the user to understand. Analysis can be performed either locally or in the clouds.

- Application: IoT applications provide an interface that the user can use to control and monitor various aspects of the IoT system.

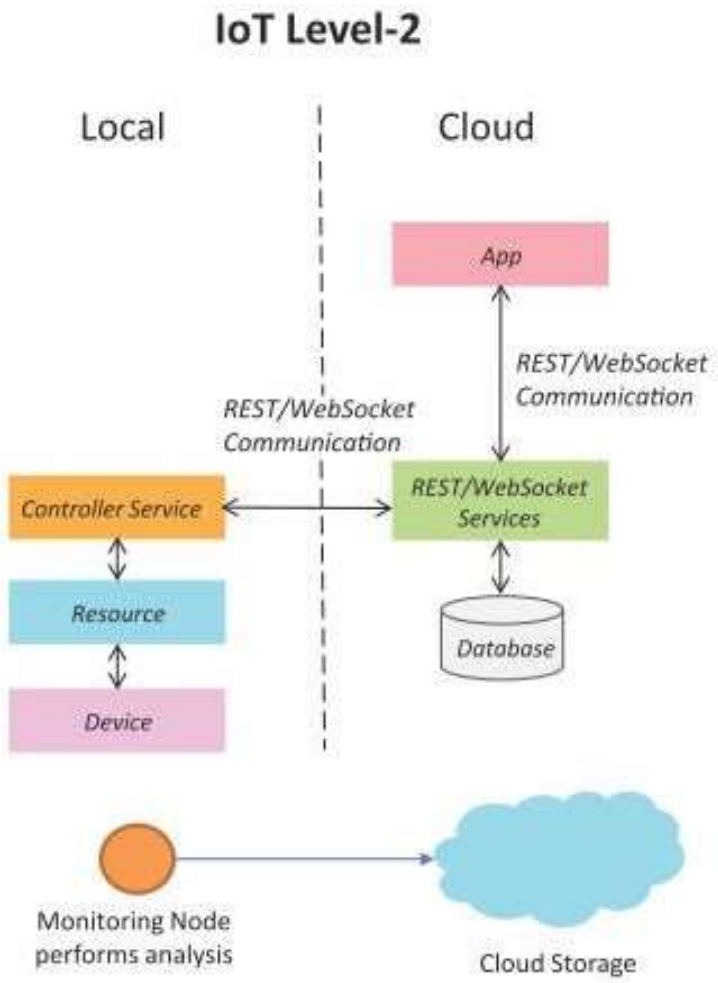
IoT Level-1

- A level-1 IoT system has a single node/device that performs sensing and/or actuation, stores data, performs analysis and hosts the application
- Level-1 IoT systems are suitable for modeling low- cost and low-complexity solutions where the data involved is not big and the analysis requirements are not computationally intensive.



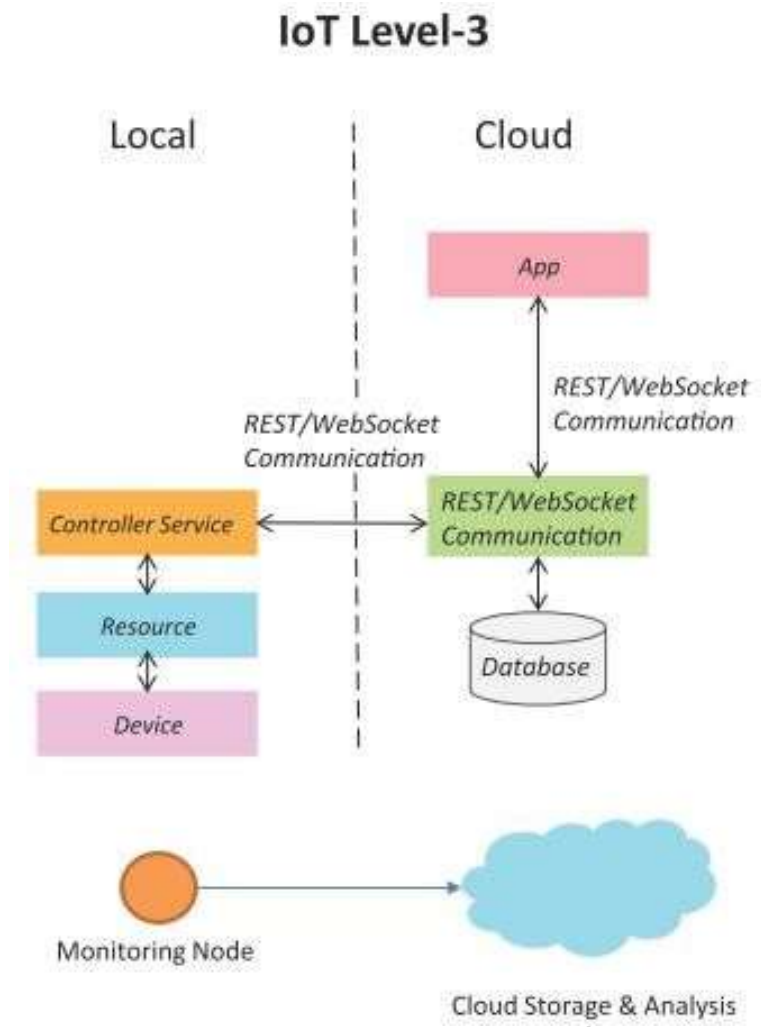
IoT Level-2

- A level-2 IoT system has a single node that performs sensing and/or actuation and local analysis.
- Data is stored in the cloud and application is usually cloud- based.
- Level-2 IoT systems are suitable for solutions where the data involved is big, however, the primary analysis requirement is not computationally intensive and can be done locally itself.



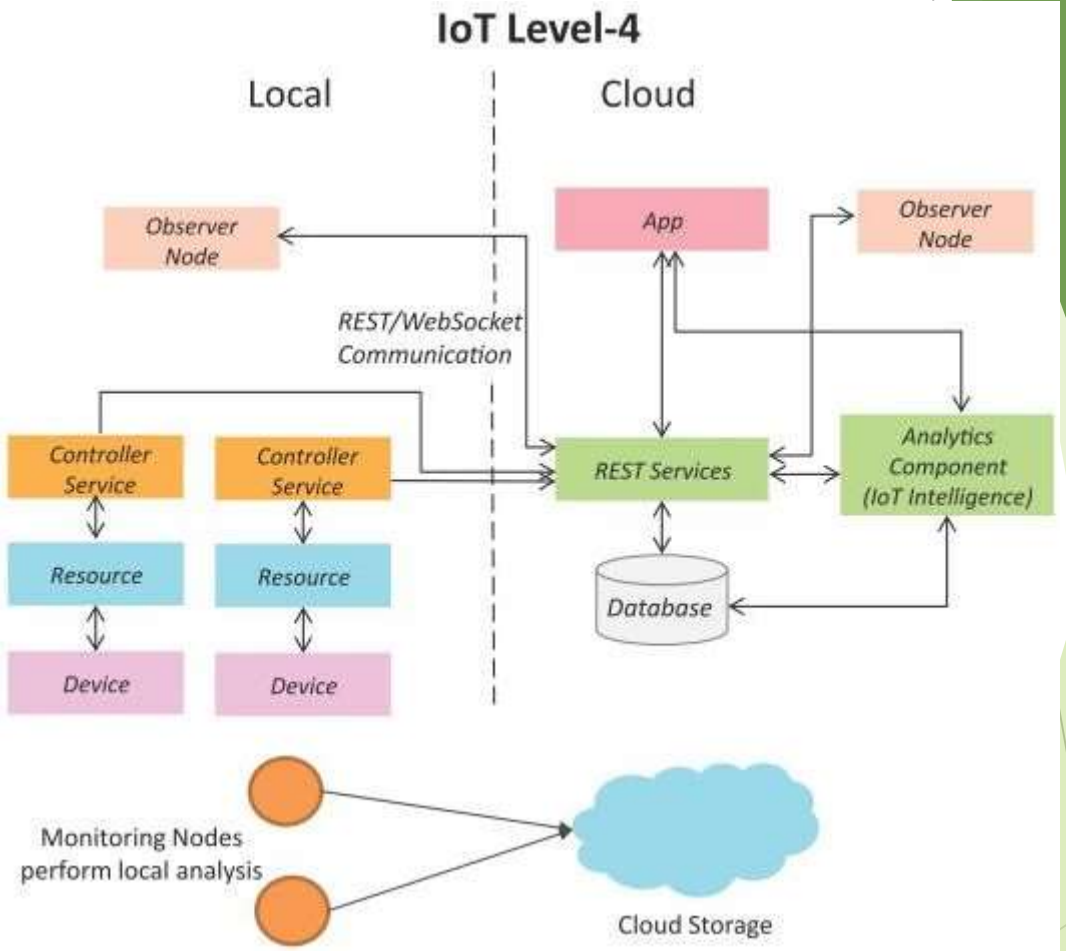
IoT Level-3

- A level-3 IoT system has a single node. Data is stored and analyzed in the cloud and application is cloud-based.
- Level-3 IoT systems are suitable for solutions where the data involved is big and the analysis requirements are computationally intensive.



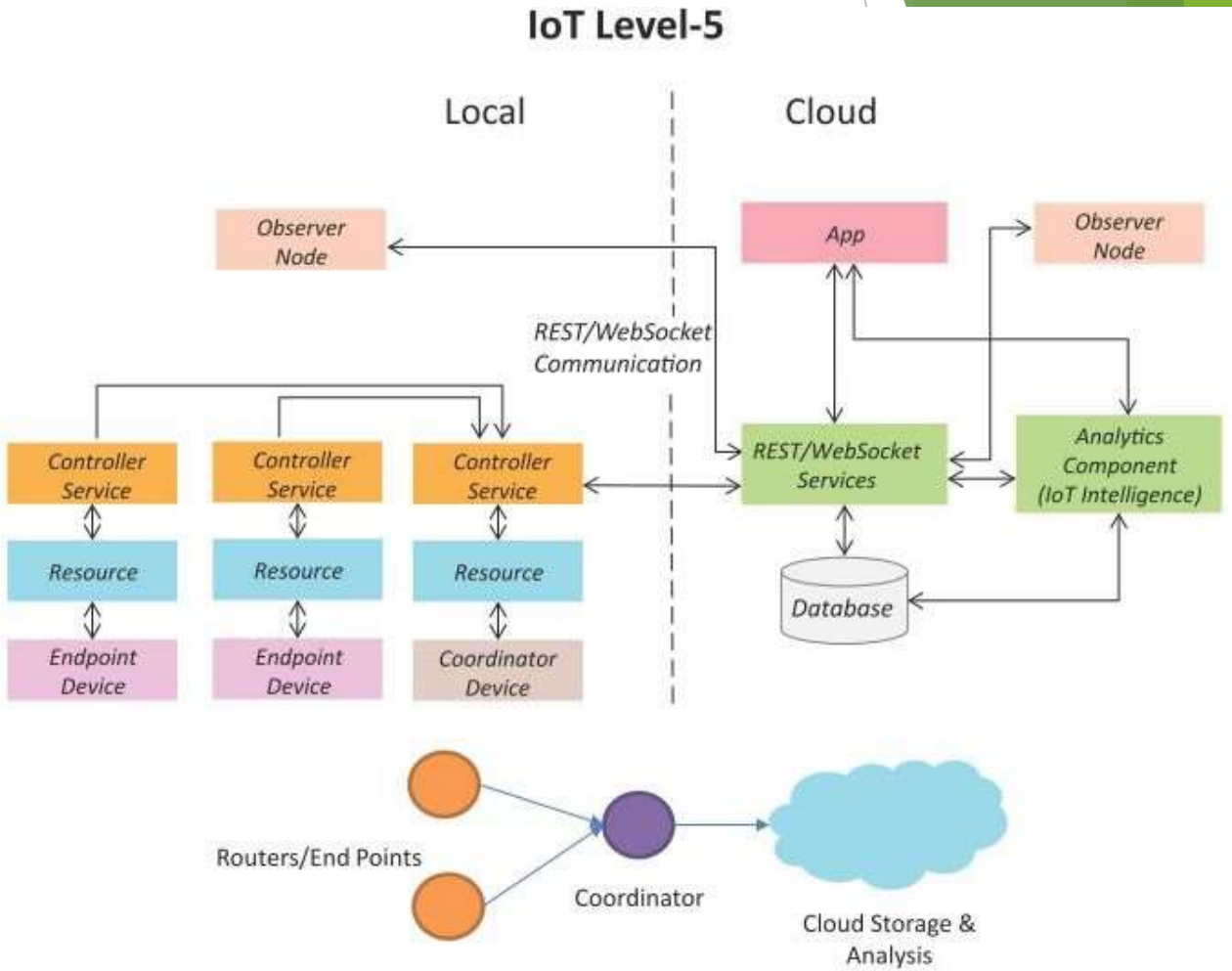
IoT Level-4

- A level-4 IoT system has multiple nodes that perform local analysis. Data is stored in the cloud and application is cloud-based.
- Level-4 contains local and cloud- based observer nodes which can subscribe to and receive information collected in the cloud from IoT devices.
- Level-4 IoT systems are suitable for solutions where multiple nodes are required, the data involved is big and the analysis requirements are computationally intensive.



IoT Level-5

- A level-5 IoT system has multiple end nodes and one coordinator node.
- The end nodes that perform sensing and/or actuation.
- Coordinator node collects data from the end nodes and sends to the cloud.
- Data is stored and analyzed in the cloud and application is cloud-based.
- Level-5 IoT systems are suitable for solutions based on wireless sensor networks, in which the data involved is big and the analysis requirements are computationally intensive.



IoT Level-6

- A level-6 IoT system has multiple independent end nodes that perform sensing and/or actuation and send data to the cloud.
- Data is stored in the cloud and application is cloud-based.
- The analytics component analyzes the data and stores the results in the cloud database.
- The results are visualized with the cloud-based application.
- The centralized controller is aware of the status of all the end nodes and sends control commands to the nodes.